24.04.95

18

PATENT OFFICE

JAPANESE GOVERNMENT

This is to certify that the annexed is a true copy of the following application as filed with this office.

Date of Application: July 30, 1993

Application Number: Patent Application No. Hei 05-189637

Applicant(s): Canon Kabushiki Kaisha

August 26, 1994

Commissioner,

Patent Office Akira Takashima

Issuance No.Hei 06-3043531

24.04.95

(Name of Document) Application for Patent

(File Number) 2589045

(Date of Submission) July 30, 1993

(Addressee) To Commissioner, Patnet Office,
Wataru Asoh

(International Patent Classification) H04N 5/232

(Title of Invention) SYSTEM CONTROL METHOD AND APPARATUS

(Number of Claims) 8

(Inventor)
(Address) c/o Canon Kabushiki Kaisha, 3-30-2, Shimomaruko, Ohta-ku, Tokyo
(Name) Akira Suga

(Inventor)
(Address) c/o Canon Kabushiki Kaisha, 3-30-2, Shimomaruko, Ohta-ku, Tokyo
(Name) Akira Mamiya

(Inventor)
(Address) c/o Canon Kabushiki Kaisha, 3-30-2, Shimomaruko, Ohta-ku, Tokyo
(Name) Fumiaki Takahashi

(Inventor)
(Address) c/o Canon Kabushiki Kaisha, 3-30-2, Shimomaruko, Ohta-ku, Tokyo
(Name) Hideaki Kawamura

(Inventor)
(Address) c/o Canon Kabushiki Kaisha, 3-30-2, Shimomaruko, Ohta-ku, Tokyo
(Name) Takashi Aizawa

(Inventor)
(Address) c/o Canon Kabushiki Kaisha, 3-30-2, Shimomaruko, Ohta-ku, Tokyo
(Name) Kenji Hatori

(Applicant for Patent)
(Identification Number) 000001007
(Postal Code) 146
(Address) 3-30-2, Shimomaruko, Ohta-ku, Tokyo
(Name) Canon Kabushiki Kaisha
(Representative) Hajime Mitarai
(Telephone Number) 03-3758-2111

(Agent)
(Identification Number) 100069877

24.04.95

(Postal Code) 146
(Address) c/o Canon Kabushiki Kaisha, 3-30-2, Shimomaruko, Ohta-ku, Tokyo
(Patent Attorney)
(Name) Giichi Marushima
(Telephone Number) 03-3758-2111

(Indication of Fee)
(Method of Payment) Deposit
(Deposit Account Number) 011224
(Amount of Payment) 21000

(List of Submitted Articles)
(Name of Article) Specification 1
(Name of Article) Drawings 1
(Name of Article) Written Abstract 1
(General Power of Attorney Number) 9003707

(Necessity of Proof) Necessary

24.04.95

(Name of Document)

Specification

(Title of the Invention)

SYSTEM CONTROL METHOD AND APPARATUS

(Scope of Claim for Patent)

(Claim 1)

A control apparatus for a control system, said system having a controller for unitarily controlling a plurality of object embeded peripheral devices through a common communication line, comprising arrangements

that said controller and said peripheral devices each has an interface of duplex configuration for transmitting or receiving data in either direction over said communication line;

that said peripheral devices each have previously stored therein data for a respective object concerning the control of functions; and

that said controller, when connected to said peripheral devices via said communication line, loads itself with said object data from said peripheral devices, then generates objects that are related to said peripheral devices and, based on said object data, displays control panels for said peripheral devices as are selectively

presented under the control of said controller,

whereby in response to activation of any items in said control panels on said controller side, said controller gives off commands which are put to said communication line through said objects to control said peripheral devices.

(Claim 2)

A system control apparatus according to claim 1, wherein said object embeded peripheral devices and said controller are provided with message switching means for transmitting or receiving messages such as control commands and data input/output commands to or from one another.

(Claim 3)

A system control apparatus according to claim 1, further including an arrangement that said object embeded peripheral devices and said controller are provided with groups of methods such as execution procedures, functions, subroutines selectively usable by said messages, and environments on which to carry out the selected methods.

(Claim 4)

A system control apparatus according to claim 1, wherein said object embeded peripheral devices and said controller are encapsulated so that the data such as those of the internal status and the variable parameters are

hidden within the inside thereof, whereby, to access said internal data, it is done indirectly by calling on the methods.

(Claim 5)

A system control apparatus according to claim 1, further including an arrangement that said object embeded peripheral devices take their own control panels and display devices in the form of graphical user interfaces which permit controls and displays of themselves to be made by a controller on the outside and have means for sending said graphic user interfaces to the controller on the outside.

(Claim 6)

A system control apparatus according to claim 5, further including an arrangement that said object embeded peripheral devices are given tables in which the corresponding ones of the methods to the operations of the control panel of said graphical user interface are defined, and have means for sending said tables to the controller on the outside.

(Claim 7)

A system control apparatus according to claim 6, further including an arrangement that said object embeded peripheral devices realize recapitulation of said graphical user interfaces and said defined method table

as objects, and have means for sending these objects to the controller on the outside.

(Claim 8)

A system control apparatus according to claim 6, further including an arrangement that said object embeded peripheral devices realize recapitulation of said graphical interfaces and said defined method tables in a predetermined descriptive language and have means for sending said descriptive language to the controller on the outside.

(Claim 9)

A system control apparatus according to claim 1, further including an arrangement that said controller has means for displaying the status of physical connections of said object embeded peripheral devices.

(Claim 10)

A system control apparatus according to claim 1, further including an arrangement that said controller has means for displaying the operative positions of said object embeded peripheral devices and means for altering the display mode.

(Detailed Description of the Invention)

(0001)

(Field of Utility on Industry)

This invention is suited to be used in controlling the system of multimedia devices which handle many items of information such as letters, voices, sound, still pictures and motion picture.

(0002)

(Prior Art)

In the field of art of audio, video, TV or like audio-visual (AV) instruments where the techniques have so far been developed in respect to analog signals at the core, the recent trend is rapidly shifting to digitization. As information of texts and still pictures in digital form is widespread, it has come to take data of texts, voices, sound, still pictures and motion picture as the so-called multimedia and handle them unitarily in a computer.

(0003)

(Subjects the invention is to Solve)

At present, however, in a case that a multimedia device (digital camera, CD-ROM player, scanner, sound board, video board or like audio input/output device or like video input/output device) is used in combination with the computer, it is necessary to install the software solely dedicated to it, called the "application" or "device driver" in the computer.

(0004)

So long as this method is relied on, therefore,

for every new multimedia device to develop, new different applications or device drivers must be prepared with different types of computers or different OS's (operating systems). Accordingly, the cost of developing the software was huge. There was another problem that an efficient and high speed control was impossible to make.

(0005)

With this method, also because that multimedia device cannot be used from another computers which were usually connected to a LAN, it was impossible to realize a concept of multimedia such that every computer could access any of the peripheral devices over the LAN.

(0006)

(Means for Solving the Subject)

The present invention has been made to solve the above-described subjects. In application to the system in which a controller unitarily controls a plurality of object embeded peripheral devices through a communication line, a characteristic feature of the invention is that the controller and the peripheral devices each have an interface of duplex configuration which transmits, or receives, data in either direction over the communication line. In every one of the peripheral devices there are previously stored data for a respective individual object concerning the control of functions. When a peripheral device is added by connecting to the communication line, the controller loads itself with the object data from that peripheral device

and forms the corresponding object to that peripheral device. At the same time, based on the object data, a control panel for that peripheral device is displayed under the control of the controller. With such a control apparatus for the system, when the user activates one of the items in the control panel on the controller side, the controller gives off a command, which is outputted through that object to the communication line, thereby controlling the added peripheral device.

(0007)

(Function)

To control the multimedia device, therefore, no special software such as application or device driver described before is needed. In addition, it is made possible to realize an environment on which the multimedia devices are shared transparent in common by another computers.

(0008)

(Embodiments)

The invention is next described in connection with embodiments thereof by reference to the drawings in detail below.

(0009)

To begin with, the system control technique used in the invention is that the multimedia devices are grasped individually as an object so the controller manages these objects unitarily.

(0010)

All the objects are managed by the controller. For this reason, every object has a function of sending its own functions and control means out to the controller. This obviates the necessary of previously preparing the control programs for the peripheral devices in the controller as was heretofore conventional. Only by connecting a peripheral device to the controller, its control can be realized.

(0011)

Again, the controller has means for displaying the above-described control means sent from the connected objects to the human being who actually gives the control commands and for letting her or him manipulate the control means. By this, the controller becomes possible to unitarily manage the multimedia devices. Also for a new multimedia device, without making any new preparation, the apparatus can cope with it. Thus, achievement of great increases of the flexibility and versatility can be realized.

(0012)

Incidentally, the object-oriented concepts used in the invention are themselves described in detail in the documents for reference, for example, Ishizuka: "Object Oriented Programming" ASCII Publishing 1988; Sakai: "Introduction to Object Oriented Technique" Ohm Co. 1990; and B.J. Cocks: "Object Oriented Programming" Toppan 1988. In the following, therefore, the embodiments of the invention are described with omission of the

fundamental techniques.

(0013)

This orientation to objects has been attracting the spotlight of attention from the standpoint of the recent trend of improving the efficiency of program development environment. Moreover, this technical idea can be widely applied even to the OS's and multimedia database. In particular, the characteristic concepts of the object orientation are:

(1) Encapsulation;

(2) Inheritance; and

(3) Messaging

With these three points as the basic concepts, the present invention has made attempts to develop and expand the technology so that it can be applied to making control of multimedia devices.

(0014)

Fig. 1 shows a logic form of connection of a multimedia controller with multimedia devices employing the object-oriented concepts of the invention. The multimedia controller 1 at the center is surrounded by a number of multimedia devices 2 with their respective individual communication paths to the multimedia controller 1 being established so that direct conversation of various items of information can be made between the multimedia controller and the multimedia devices in one to one basis. It will be appreciated that the control is made by transmitting messages to each other over that

communication path. The term "multimedia devices" herein used means, specifically speaking, CD players, digital VTRs, digital cameras, digital TV sets and other AV devices, and digital FAX, digital copiers, printers and other OA devices, that is, all of those devices which deal with multimedia data.

(0015)

The controller is assumed here to be part of hardware dedicated solely to this purpose. But it is also possible to realize an equivalent controller by installing an especial OS and a particular application on the commonly available processor in the personal computer or word processor.

(0016)

Referring next to Fig. 2, there are shown three configurations (a) to (c) for physically connecting a multimedia controller to a number of multimedia devices to establish the respective duplex communication paths.

(0017)

The daisy chain line of Fig. 2(a) is employed in SCSI bus (ANSI X3.131-1986). The star configuration of Fig. 2(b) is employed in Ethernet (IEEE 802.3) 10BaseT. The multipoint line of Fig. 2(c) is employed in Ethernet 10Base2/5.

(0018)

It is also to be noted that with regard to another possible configurations, there is GPIB (IEEE 4888) as obtained by mixing the (a) to (c). Even in Ethernet,

the (b) and (c) may be mixed. With regard to another possible communication systems, there are optical fiber cables and ISDN. So it is to be understood that, besides those of Fig. 2, many other combinations are possible to make and may be selectively employed as desired.

(0019)

How to establish such duplex communication paths and which to select are not essential to the invention, so no particular remarks are given except that, as the communication system differs from one to another, some physical limitations are laid on the transfer speed, the number of connected devices, the length of the cable, the shape of the connector, etc. For the each other's transmission of messages, because the protocols have their hierarchies differentiated from each other, these limitations are out of question. In order to insure that the peripheral devices each are connected to the controller reliably and accurately, however, there is need to provide for the system with at least one physically (mechanically and/or electrically) common interface.

(0020)

To realize high speed transmission of data such as those of motion pictures, it is recommended to employ what is faster than Ethernet, that is, the optical transmission such as FDDI (Fiber Distributed Data Interface) or B-ISDN. But, in here, for the purpose of simplicity of explanation, discussions are conducted on

assumption that Ethernet 10Base2(/T) which, because of its cheap price, is widely used is adopted as the common communication connector.

(0021)

The internal pieces of hardware of the usual multimedia device are shown in an block diagram of Fig. 3.

(0022)

A plurality of multimedia devices are connected via a LAN 4 to the controller. Now this LAN is Ethernet so each of these devices is provided with an interface 20 for implementing its protocol (TCP/IP). This can be realized by using an exclusive LSI or the like. It is in here that the transmitted message itself is taken out. Conversely it is from here that a message is sent out to the controller. An example of these messages, if in the Objective-C, is given, as the general format, by the following expression:

(0023)

(Terminal Object; Method Name; Parameter)

In another languages, the expression takes different styles, but is similar in the following basic components:

(0024)

(1) Addressing a terminal object;

(2) Selecting a method (instruction to execute); and

(3) Putting data in parameters, if any.

How to deal with this message is described in connection with the flow of software of Fig. 33.

(0025)

In the interior of the multimedia device, a CPU 11 processes all software and controls all hardware through an external bus 10. The programs, the initial values and proper data are stored in a ROM 12. To temporarily store data and internal parameters such as those representing the device status, there is a RAM 13. When executing the programs, this RAM 13 is used as a work area. A data I/O 14 is used in accessing multimedia data stored on an internal or external medium 15. A mechanical system driving portion 16 controls mechanical parts 17 such as an electric motor. An electrical system driving portion 18 controls electrical circuits for switches SW and indicators such as LEDs. As the multimedia data are of digital form and range from pictures to sound to texts, it is possible for the medium 15 to take various types, namely, optical disks such as CD-ROM and MD, magnetic tapes such as DCC and DAT, and semiconductor memory cards.

(0026)

Referring next to Fig. 4, a block diagram shows the internal hardware aspect of the multimedia controller 1. Connections to the multimedia devices are established via the LAN 4. Now this LAN 4 is Ethernet, so there is an interface 31 for implementing its

communication protocol (TCP/IP). This can be realized by using a LSI or the like solely dedicated thereto. It is in here that the transmitted message itself is taken out. Conversely it is from here that messages are sent out to the multimedia devices.

(0027)

In the interior of the multimedia controller 1, a CPU 21 processes all software and controls all hardware through an external bus 30. The programs, the initial values and proper data are stored in a ROM 22. To temporarily store data and internal parameters such as those representing the device status, there is a RAM 23. When executing the programs, this RAM 23 is used as a work area. A multimedia filing device 25 performs storing, retrieving, reproducing or editing of multimedia data, regardless of whether filing device is an internal or external medium. Accessing to it is controlled by a data I/O 24. An electrical system driving portion 28 controls electric circuits for switches SW and LEDs or like indicators. A display 27 constitutes a man-machine interface. Its displaying operation is controlled by a controller 26. There is further included a mouse or like pointing device, though not shown.

(0028)

Fig. 6 is a diagram of the hierarchy of systems in the software aspect of the multimedia device. The internal block diagram of Fig. 3 refers to hardware 57. An OS 58 is fundamentally in charge of this hardware.

What type of OS to select is not itself particularly limited, but it is desired that the real-time facility and the multitasking capabilities that run more than one program in parallel at a time are available in combination. On this OS, the multimedia device has a class library 59 which is differentiated from the other devices in order to realize embedding of an object into that multimedia device.

(0029)

Though not shown, the multimedia device has another library concerning the control panel for itself and the control program. When connected to the multimedia controller, this library is transferred thereto, thus permitting a specific control to the multimedia device to be made at the side of the controller. There is also a C function 60 to be used as a timer and in performing arithmetic computations.

(0030)

At the top of the hierarchy, there is an application software 61 which takes its part in controlling the main system of the multimedia device and communicating with the multimedia controller, and as the user interface. The use of this application makes it possible that the main system of the multimedia device as one object is controlled in a variety of ways by transferring messages from or to the controller, and that the internal parameters are read as instance variables and altered.

(0031)

Fig. 5 is a diagram of the hierarchy of systems in the software aspect of the multimedia controller. The internal block diagram of Fig. 4 refers to hardware 50. An OS 51 is fundamentally in charge of this hardware. In here, too, what type of OS to select is itself not particularly limited. But it is desired that the real-time facility and the multitasking capabilities are available in combination.

(0032)

On this OS there is a window server 52 which is in charge of the whole of a GUI (Graphical User Interface) that displays a plurality of control panels for the connected multimedia devices and the status of all links in the network on the screen, and controls coordination of inputting and outputting of data. A common class library 53 stores what has been made ready beforehand in the controller by itself, that is, a set of basic and common components (in the form of objects) concerning the user interface, namely, buttons, slide volumes and text presentation areas and also concerning the control.

(0033)

For every multimedia device, on the other hand, a specific class library 55 stores a set of components (in the form of objects) concerning the unique panel display and control. This specific library, as described before, increases its content each time one more

multimedia device is brought into connection with the system, as an additional set is sent from that device. This procedure will be described later in more detail. There is also a C function 54 for the timer and arithmetic computation. At the top of the hierarchy, there is an application software 56 which undertakes in controlling coordination of all the connected multimedia devices and communicating with these multimedia devices and that functions as the user interface.

(0034)

The flow of control signals and the transmission of messages between this controller and the multimedia device are described below.

(0035)

Fig. 7 shows a state of the system before the multimedia device is connected to the multimedia controller. In Fig. 7, digital data are transmitted over a communication line or LAN 4. A multimedia controller 1 controls operations of all parts of the system. A multimedia device 2 to be connected to the LAN 4 is shown with its structure of construction in generalized form. 205 is the one of software objects (hereinafter abbreviated to "objects") which always resides in the multimedia controller 1 and coordinates all parts of the system, named "system director".

(0036)

1064 is an object which functions as a multimedia device and which is differentiated from the

other objects on the LAN 4, named "multimedia device". This object is further comprised of three objects 1065, 1066 and 1067.

(0037)

The "controller for multimedia device" object 1065 is in charge of hardware to realize a majority of functions of the multimedia device 2. The "data input to multimedia device" object 1066 is used for entering digital data as transmitted from the other devices over the LAN 4. The "data output from multimedia device" object 1067 is used for transmitting digital data to the other devices over the LAN 4.

(0038)

When the multimedia device 2 is connected to the multimedia controller 1 via the LAN 4, an object that stands in the place of that multimedia device 2 must be formed in the multimedia controller 1. To describe this "deputy multimedia device" object, a specification is written in a file 1061. This file comprises a section 1062 for the specification of a control panel for the multimedia device 2 and another section 1063 for the specification of an input or output of data to or from the multimedia device. In particular, the section 1062 for describing the "control panel for the multimedia device" object realizes the function of describing the control panel so that one interacts with the multimedia device 2 by means of a GUI, that is, the function of the language in which to describe the GUI.

(0039)

Fig. 8 is a diagram to explain the state of the system after the multimedia device 2 is connected to the LAN 4. In Fig. 8, what is now formed in the multimedia controller 1 is an object 1068. In the interior of the multimedia controller 1, this object acts as substitute for that multimedia device 2, so it is called "deputy" multimedia device. This object 1068 comprises an object 1069 which functions as the control panel for the multimedia device 2, named "control panel for multimedia device", another object 1070 which, when to input data, functions as substitute for the "data input to multimedia device" object 1066, called "deputy" object, and still another object 1071 which functions also as substitute for the "data output from multimedia device" object 1067, called "deputy" object.

(0040)

Fig. 9 is a diagram of the structure of the common class library. For the objects having a similar feature, their common attribute and function have to be defined. To this purpose, a class, say a first class 1079, functions as a template. The first class 1079 to the p-th class 1085, totaling p classes, are summed up in a library 1086. This is conventionally called "class" library. All the objects belong to the respective specified classes. The type and name of data for the internal variables and the type and name of data for the internal functions representing the data processing means

(usually called "class methods" the one of the objects which belongs to a particular one of the classes should possess are defined in a portion 1080. To allow one to access to the class methods, all the codes of the class methods are cited in the form of a table 1081 relative to a pointer. A first function code 1083 through a k-th function code 1084, totaling k function codes for the k class methods, are stored in a "code" portion 1082.

(0041)

Fig. 10 is a diagram of the structure of a typical object. In Fig. 10, an object 234 comprises a portion 244 for accommodating the pointer that is to go to the class method table, communicating means 245 for messages, processing and retrieving means 246, a portion 239 for methods and a portion 235 for internal data. The "method" portion 239 comprises beginning with first data processing means 240, second data processing means 241 and so on and terminating at m-th data processing means 242, totaling m data processing means. The portion 235 for internal data comprises beginning with a first internal data 236, a second internal data 237 and so on and terminating at an n-th internal data 238, totaling n internal data.

(0042)

All the internal data in the portion 235 differ with the different objects and, therefore, are left as they stand in the interior of each of the objects. The data processing means in the "method" portion 239, on

the other hand, can be used in common by the other objects, if of the same class. To assure this, therefore, a class method table 243 is provided so that the first data processing means 240 through the m-th data processing means 242 are made manageable by each of the classes. Thus, such common data processing means are shared by a number of objects which belong to the same class. To look up the class method table 243 from every object, the pointer is brought into this table from an accommodating portion 244 therefor.

(0043)

The message communicating means 245, when in receiving a message from another object, transfers it to the processing and retrieving means 246, where the message is analyzed to identify its address section and the corresponding one of the data processing means to it is retrieved from the "method" portion 239 (virtually the class method table 243). As the selected data processing means operates, the data section of the message, the internal data from the portion 235 and some external data are processed in a predetermined way. In some case, such processing will result in production of a message. If so, this message is transmitted from the communicating means 245 to that other object.

(0044)

Fig. 11 is a diagram of the structure of the system director object 205. A space 1072 accommodates a pointer to a class method table 1073. 1047 is an object

forming means for producing the deputy multimedia device object 1068 based on the information from the file 1061. Checking means 343 determines whether or not the formats of input or output data between the objects are compatible with each other. Another object forming means 380 produces various application objects of different aims. Further included are communicating means 1074 for messages, processing and retrieving means 342, a "method" storage part 1075 and an "internal data" storage part 1076. The internal data are an object ID 1077, control data 344 to be used in establishing links between any two of the multimedia devices when to carry out certain operations, and object record data concerning the connected multimedia devices or the produced objects therefor.

(0045)

When a multimedia device 2 is connected to the LAN 4, using the deputy multimedia device object forming means 1047, the system director object 205 reads the file 1061 for describing that object. From the information obtained from this file 1061, it then determines which class the object that should be produced belong to. Based on the definition in the portion 1080 of the corresponding class in the class library 1081, the deputy multimedia device object 1068 is made up.

(0046)

Fig. 12 is a diagram of the structure of the "control panel" written section of the "deputy object"

written file. In Fig. 12, the section 247 for describing a control panel object comprises first to i-th databases 248 to 249 to be used in describing i articles of the object. One database consists of data 250 for recognizing the object, data 254 for drawing the object and data 260 for object linkage.

(0047)

The data 250 for recognizing the object are of a name 251 of the class which the object belongs to, a unique ID 252 to the i-th article, and an ID 235 of the object to which the i-th article is appendant.

(0048)

The data 254 for drawing the object are used to depict a button or like object constituting part of the control panel window 231 on the screen, comprising first to j-th packs 255 to 259 of data for drawing j articles of the object, One pack consists of data 256 for the location and size at and to which to depict the object, data 257 for the pattern and color and an object image 258.

(0049)

The object link data 261 provide information about the links in the network and are used when one of the items constituting the control panel, for example, the controller object 207, is to connect to another, say relational object, comprising first to k-th data 261 to 264 for establishing k links between the items. One data are of an ID 262 of a relational object, and an message

263 for transmission to the relational object.

(0050)

Fig. 13 is a diagram of the structure of a "deputy data I/O object" section in the "deputy object" file. In Fig. 13, the "deputy data I/O object" section 650 includes first to m-th data 651 to 652 for producing m articles of a deputy input object. Each data are of an ID 652 of its own article, an ID 653 of a link terminal "data input" object and a compatible file type list 654. The section 650 further includes first to n-th data 659 to 663 for n articles of a deputy "output" object. Each data are of an ID 660 of its own article, an ID 661 of a relational object, and a compatible file type list 662.

(0051)

Next, taking an example of a digital VTR to which the above-described system control method of the invention is applied, the operation of the control system for multimedia devices 2 is described below.

(0052)

Fig. 14 is a diagram showing the state of the control system before an object embedded digital VTR is connected to the multimedia controller. In Fig. 14, as an object is embedded into the digital VTR 203, this "digital VTR" object 206 always resides in the digital VTR 203 and functions as the object embedded digital VTR as viewed from the other devices on the LAN. The digital VTR object 206 is further constructed from three objects.

Of these, a "digital VTR controller" object 207 controls the hardware of the digital VTR 203.

(0053)

Another or "data input to digital VTR" object 208 is used to input digital data as transmitted from the other devices over the LAN 4. The other or "data output from digital VTR" object 209 is used to output digital data for transmission to the other devices over the LAN. When the digital VTR 203 is connected to the multimedia controller 1 via the LAN 4, a deputy digital VTR object is generated in the interior of the multimedia controller 1 based on the information from a "deputy digital VTR object" written file 210.

(0054)

This file 210 comprises a section 211 in which the specification of a control panel for the digital VTR 203 is written, or which is used in describing a "control panel for the digital VTR" object and another section 212 in which the specification of an object that acts as substitute for the data input/output of the VTR 203, or which is used in describing the "deputy data I/O of digital VTR" object.

(0055)

Fig. 15 is a diagram of the structure of the VTR controller object 207, where a portion 1009 accommodates a pointer that goes to a class method table 1018. This table is formed with a wide variety of data processing means including reproducing means 1019

for operating the play mode under the control of the hardware of the VTR and recording means 1020. 1010 is communicating means for messages. 1011 is processing and retrieving means. Though a "method" part is shown at 1012, it is in actual practice that the data processing means are presented by a class method table 1018. The internal data in a portion 1015 are of many variables and status information necessary to control the digital VTR, for example, the tape running speed 1016 and the current tape footage 1017.

(0056)

Let us first explain an operation which occurs when the digital VTR 203 is connected to the LAN 4. Fig. 16 is a flowchart of the routine for this operation. Fig. 17 is a plan view of a window for the multimedia controller 1 on the screen. In Fig. 17, the multimedia controller window 228 contains a number of icons of which the icon 229 appears when the digital VTR 203 is connected to the LAN 4. Using a mouse or like pointing device, one can choose a location with a cursor 230. The pointing device, though not shown, is provided with buttons. The user presses the button and then releases it. This pressing once is usually called clicking. Pressing it twice in quick succession is called double clicking. Incidentally, as other usable devices, mention may be made of a camera (for inputting still pictures), a tuner, a television set, various relational databases and a CD. To allow the user to select these options, the window 228

displays their icons.

(0057)

Fig. 18 is a diagram to explain the state of the system when the object embedded digital VTR 203 as an example of the multimedia device is connected to the LAN 4. In Fig. 18, an object 220 is now formed in the interior of the multimedia controller 1. This object serves as substitute for the digital VTR 203, being named "deputy digital VTR". The deputy digital VTR object 220 is constructed from a deputy "control panel for digital VTR" object 221 which functions as a control panel for the digital VTR 203, another or deputy "data input to digital VTR" object 222 which, when inputting data, functions as substitute for the data input object 208, and yet another or deputy "data output from digital VTR" object 223 which functions also as substitute for the data output object 209.

(0058)

Referring now to Fig. 16 and Fig. 17, the routine for such an operation is described below. When the digital VTR 203 is connected to the LAN (636), the system director object 205 detects establishment of a new connection (637). Then it sends a device ID to the digital VTR 203 (638).

(0059)

Using the deputy multimedia device object forming means 1047, the system director object 205 then loads the "deputy digital VTR object" file 210 from the

digital VTR 203 (639). Based on the information from the file 210, the system director object 205 then generates a deputy digital VTR object 220 in the interior of the multimedia controller (640) by using the deputy multimedia device object forming means 1047. Such a procedure results in a change of the status of the network as shown in Fig. 18. Then, the deputy digital VTR object 220 presents the display of an icon 229 for the digital VTR 203 in the multimedia controller window 228 (641). After this, the system stands by for instructions from the user (642).

(0060)

Subsequently, with the help of the window displayed on the screen based on the "control panel for digital VTR" object of the multimedia controller, the user will activate the digital VTR. On this, the control system can operate the digital VTR through the intermediary of the deputy digital VTR object 220 in the multimedia controller 1.

(0061)

Next, the relationship between the content of the "deputy digital VTR object" written file 210 and the object to be generated is described in detail below.

(0062)

Fig. 19 shows an icon for the VTR 203 and Fig. 20 shows an example of the control panel window on the screen. This icon 229 of Fig. 19 appears when the digital VTR 203 is connected to the LAN 4. The digital

VTR control panel object 221 depicts a window of Fig. 20 by default on the screen. In this window, there is an option menu 232 for selectively display the control panel windows on the screen. As the tape is running, the passed time is displayed in a time counter box 265. As the digital VTR 203 has a number of control modes, there is a mode option box 267 containing a first switch button for setting a control mode by default and a second switch button 268 for selectively setting more elaborate control modes. A rewind button 269, a reverse play button 270, a pause button 271, a play button 272, a fast feed button 273, a stop button 274 and a record button 275 are displayed in array.

(0063)

Fig. 21 is a diagram, partly in pictorial form, to explain the correspondences between the classes the objects belong to and the constituent elements of the "control panel for digital VTR" object. All classes to which the fundamental constituent elements belong are defined previously in the class library 1081. This library is kept in the multimedia controller 1. As is obvious from Fig. 21, all the constituent elements of the "control panel for digital VTR" object 221 function as respective individual objects constituting the "control panel for digital VTR" object 221.

(0064)

In Fig. 21, the frame of a control panel window 231 on the screen corresponds to the VTR control

panel object 221 (ID = 1) of panel class. The control panel display option menu 232 corresponds to the panel view setting menu object 285 (ID = 2) of menu class. The time counter box 265 corresponds to the time counter object 286 (ID = 3) of form class. The rewind button 269 corresponds to the rewind button object 287 (ID = 4) of button class. The reverse play button 270 corresponds to the reverse play button object 288 (ID = 5) of button class. The pause button 271 corresponds to the pause button object 289 (ID = 6) of button class. The play button 272 corresponds to the play button object 290 (ID = 7) of button class. The fast feed button 273 corresponds to the fast feed button object 291 (ID = 8) of button class. The stop button 274 corresponds to the stop button object 292 (ID = 9) of button class. The record button 275 corresponds to the record button object 293 (ID = 10) of button class.

(0065)

The control mode dialog box 266 corresponds to the control mode selection object 294 (ID = 11) of button group class. The first switch button 267 corresponds to the default button object 295 (ID = 12) of radio button class. The second switch button 268 corresponds to the advanced button object 269 (ID = 13) of radio button class.

(0066)

Next, of the objects constituting the "control panel for digital VTR" object 221 shown in Fig. 21, an

example of the play button is taken to explain how to make up an object 290 therefor.

(0067)

Fig. 22 is a diagram to explain the formation of the play button object 290. In Fig. 22, there are shown elements 297, 298, 299, 300, 601, 602, 603, 604, 605, 606, 607, 608, 609, 610 and 611 written in the "control panel object" section of the "deputy digital VTR object" file 210.

(0068)

A piece of information 297 for recognizing the object are formed with a class name 298, an object ID 299 and a superior object ID 300. A first piece of information 601 for drawing an article of the object are formed with data 602 for location and size, data 603 for pattern and color and an object image 604. A second piece of information 605 for drawing an article of the object are formed with data 606 for location and size, data 607 for pattern and color and an object image 608. Object link data 609 consist of a link terminal object ID 610 and an outgoing message 611.

(0069)

The play button object 290 of button class is generated by information from that class and the "'control panel of objects' object" section 247 in the "deputy digital VTR object" file 210. A portion 613 accommodates a pointer that goes to a class method table 625, where it points to methods in the button class.

The button class method table 625 is formed from means 626 responsive to start of generation of an object of button class for initializing the internal variables of the button object are initialized, means 627 for depicting the button object to display, and click response means 628. The user moves the mouse or like pointing device to position the cursor 230 on top of the play button. Responsive to clicking on this button, the click response means 628 changes the display of the button for a moment to inform the user of the fact that the button object has been activated and sends a message to another object.

(0070)

The definition of every data processing means in the method table for these button classes is described in each class. Therefore, not only the play button object 290 but also all the other objects which belong to the button class share the common button class method table. 614 is message communicating means; 615 is processing and retrieving means; 616 is a portion for methods; and 620 is a portion for internal data. The internal data are of an object ID 621, the state 622 of the button, drawing parameters 623 and link data 624. The one of the types of the internal data which not only the play button object 290 but also all the other button objects which belong to the button class should possess is described in the class.

(0071)

The system director object 205 first reads

in the " deputy digital VTR object" file 210, when to make up any of the objects. In the example of Fig. 22, it then accesses the data for recognizing the object and, on the basis of the description of its class name 298, forms the objects of the button class. For the play button object 290 to generate, the system director object 205 cooperates with the button initializing means 626 to initialize the "internal data" portion 620. According to the example of Fig. 22, the object ID is set to ID = 7 by the description of the object ID 299. From the description of the superior object ID 300, the system director object 205 recognizes that the play button object 290 belongs to the "control panel for digital VTR" object 221. In such a manner, based on the information from the object to any of the objects of principal interest, the system director object 205 recognizes which one of the objects contains the other. Thus, a number of the constituent objects are put together to form a complex object.

(0072)

The button drawing means 627 depicts the play button object 290 on the basis of the drawing parameters 623 and the data 622 of the state of the button. The button drawing means 627 is automatically activated when the button object is generated and when the superior object moves.

(0073)

The first piece of information 601 describes

that article 625 of the button which is effective in the situation when it is not pushed. The data 602 for location and size are used to determine a location at which the play button object 290 is depicted in the window for the "control panel for digital VTR" object 221 and a size to which a rectangular frame of the play button 625 is limited. To figure the rectangular frame, as the button is not pressed, the data for drawing the button article 625 are expressed in the coordinates of the "control panel for digital VTR" object 221. Suppose, for example, the left hand upper and right hand lower corners are taken into account, then the data of that rectangular frame have a form like (X1, Y1) and (X2, Y2). For the play button object, when not pressed, its pattern and color are determined based on information either from the pattern and color data 603 or from the object image 604. The data 603 for pattern and color are described in an appropriate language to draw lines and paint colors, that is, to describe the object in a graphical form. The object image 604 is expressed in the form of bit map data. In general, the expression by the former costs a less amount of data, but the latter has rather a high degree of freedom.

(0074)

The second piece of information 605, similarly to the first piece of information 601, describes that article 626 of the button which is effective in the situation when it is pressed. Based on both of the first

data 601 and the second data 605, the values of the drawing parameters 623 are determined. The link data 624 are set in based on information from the data 609 for object linkage. Therefore, as the message to transmit, "play" is set in and, as the link terminal object ID, a link terminal object ID is set in. In connection with the latter, it is to be noted that when to transmit the message, only one of the terminal objects over the entirety of the system should be selected to receive this message. To this purpose, the link terminal object ID to be used is set in the preceded form by the device ID the system director object 205 has assigned to the digital VTR, when the digital VTR 203 was connected to the LAN 4.

(0075)

Even if it happens that two devices have their link terminal objects to use the same ID, therefore, the message can be transmitted right to the desired object. The button state data holds the information of whether or not the button is pressed.

(0076)

Fig. 23 is two flowcharts, one of which shows the operation when the user positions the cursor 229 on top of the icon 229 of the digital VTR 203 and double clicks, and another one which shows the operation when the user has manipulated the control panel.

(0077)

Fig. 24 is a plan view of the display of a

window for the multimedia controller 1 on the screen as presented when the user has double clicked on the icon 229 of the digital VTR. In Fig. 24, the control panel window 231 for the VTR 203 is selected by default, and the play button is shown at 272.

(0078)

Fig. 29 is a diagram of the relationship between the structure of the "control panel for digital VTR" object of panel class and the object description data.

(0079)

In Fig. 29, a portion 1401 accommodates a pointer that goes to the class method table, in this instance, a panel class method table 1402. This table is formed with panel initializing means 1403 for initializing the panel object, panel drawing means 1404 for showing the panel in a graphical form, and click response means 1405 for activating the clicked object. Message communicating means 1406, processing and retrieving means 1407, and an "internal data" part 1410 are shown. The internal data are of an object ID 1411, the panel state 1412, and drawing parameters 1413. The "internal data" part 1410 is initialized according to the description of the "deputy digital VTR object" file 210. The "'control panel for digital VTR' object" section 211 in this file 210 comprises data 1414 for recognizing the object, a first pack 1418 of data for drawing an icon 1426 of the digital VTR 203, and a second pack 1422 of data for drawing the frame of the control

panel window for the digital VTR. The data 1414 for recognizing the object are of a class name 1415 (panel class), an object ID 1416 (ID = 1), and a superior object ID 2417. The first data pack 1418 consists of data 1419 for location and size, data 1420 for pattern and color, and an object image 1421. The second data pack 1422 consists of data 1423 for location and size, data 1424 for pattern and color and an object image 1425.

(0080)

Referring to Fig. 23 and Fig. 29, the routines for displaying the control panel window for the digital VTR 203 and activating the play mode are described below. As described in connection with the routine of Fig. 16, at a time when the system director object 205 has generated the deputy digital VTR object 220, the deputy digital VTR object 220 presents the display of the icon 229 as obtained based on the icon image 1426. For now, when the user double clicks on the icon 229 for the digital VTR (643), the control panel object 221 of the deputy digital VTR object 22 sends a message of executing the drawing function to all objects constituting the control panel object 221. Based on this message, all the objects shown in Fig. 21 activate the drawing means. Meanwhile, the control panel object 221 depicts the frame of the control panel window for the digital VTR based on the second pack of object drawing data. As a result, the digital VTR control panel window 231 is displayed on the screen (644) as shown in Fig. 24 and waits for

instructions from the user (645). With this, when the user positions the cursor 230 on top of the play button 272 and clicks (646), the control panel object 221 sends a message "PLAY" to the controller object 214 of the digital VTR 203 (647). Responsive to this message, the controller object 214 of the digital VTR 203 activates the play executing means (648), thus starting an operation of the play mode of the digital VTR 203.

(0081)

As has been described above, according to the invention, when a multimedia device is only connected to the multimedia controller via the LAN, its object necessary to coordinate the multimedia device with the others, or deputy multimedia device object, is automatically generated in the multimedia controller. Further, the control panel necessary to choose the multimedia devices is automatically displayed in the multimedia controller window on the screen. With the help of this control panel, the user activates an item. Then an unique message is transmitted to the controller object of the corresponding multimedia device. So the desired functions are executed. Since the information necessary to generate the deputy multimedia object, which in turn is necessary to manipulate the multimedia device, is obtained from the "deputy multimedia device object" written file as read from the multimedia device, what suffices for the multimedia controller is only the fundamental class library. So there is no need to store

the related database to any specific multimedia device in advance.

(0082)

Fig. 25 is a diagram of the relationship between the structure of the deputy "data input to digital VTR" object and the data for describing the object. In Fig. 25, the deputy "data input to digital VTR" object 222 contains a portion 668 for accommodating a pointer that goes to the class method table, in this instance, a deputy data input class method table 679. This table is formed with means 680 for initializing the deputy data input object, means 681 for updating the link data and compatible file type reply means 678.

(0083)

669 is message communicating means; 670 is processing and retrieving means; 671 is a "method" part; and 674 is an "internal data" part. The internal data are of an object ID 675, another ID 676 which represents the related data input object, compatible file types 677 and links 1006 with data output objects.

(0084)

The "deputy digital VTR object" file 210 contains a "deputy 'data input/output of digital VTR' object" section 212. Based on the information from this section, the deputy "data input to digital VTR" object is generated. The data for the "deputy input" object described in the section 212 are of an object ID 683 (in this instance, ID = 1), a related data input object ID

(in this instance ID = 1), and a list 685 of compatible file types (in this instance, assumed to be formats so called "AV1" and "AV2"). According to the description of these parameters, the deputy input object initializing means 680 initializes the "internal data" portion 674.

(0085)

Fig. 26 is a diagram of the relationship between the structure of the deputy "data output from digital VTR" object and the data for describing the object. In Fig. 26, the deputy "data output from digital VTR" object 223 contains a portion 690 for accommodating a pointer that goes to the class method table, in this instance, a deputy data output class method table 1048. This table is formed with means 694 for initializing the deputy data output object, means 695 for updating the link data and compatible file type reply means 700.

(0086)

691 is message communicating means; 692 is processing and retrieving means; 693 is a "method" part; and 696 is an "internal data" part. The internal data are of an object ID 697, another ID 698 which represents the related data output object, compatible file types 699 and links 688 with the data output object.

(0087)

Based on the information from the "deputy digital VTR object" file 210 at the "deputy 'data output from digital VTR' object" section, the deputy "data output from digital VTR" object is generated. For now, the

data 1001 for the deputy data output object described in the section 212 are of an object ID 1002 (in this instance, ID = 1), a related data output object ID 1003 (in this instance ID = 1) and a list 1004 of compatible file types (in this instance, assumed to be formats so called "AV1" and "AV2"). According to the description of these parameters, the deputy data output object initializing means 694 initializes the "internal data" portion 696.

(0088)

Fig. 27 is a diagram of the structure of the "data input to digital VTR" object. This object 208 includes a portion 1030 for accommodating a pointer that goes to the class method table, in this instance, a data input class method table 1031. This table 1031 is formed with file writing means 1032, data receiving means 1033 and link data updating means 686. 1023 is message communicating means; 1024 is processing and retrieving means; 1025 is a "method" part; 1028 is an "internal data" part; 1029 is an object ID; and 1030 is link data.

(0089)

Fig. 28 is a diagram of the structure of the "data output from digital VTR" object. This object 209 contains a portion 1035 for accommodating a pointer that goes to the class method table, in this instance, a data output class method table 1044. This table is formed with file reading means 1045, data transmitting means 1046, and link data updating means 687. 1036 is

message communicating means; 1037 is processing and retrieving means; 1038 is a "method" part; 1041 is an "internal data" part; 1042 is an object ID; and 1043 is link data.

(0090)

After the deputy data input object 222 and deputy data output object 223 of the digital VTR 203 have been generated in the multimedia controller, these objects function as if they were chief ones, or the "data input to digital VTR" object 208 and the "data output from digital VTR" object 209. Now suppose the digital VTR receives, for example, a file from another multimedia device by the copy function, then the system director object 205 inquires of the deputy "data input to digital VTR" object 222 what types of files are possible to input. Responsive to this inquiry, the compatible file type reply means of the deputy "data input to digital VTR" object 222 gives off information about the file types the digital VTR 203 can accept.

(0091)

If the type of the file to be copied is found to be present among them, a link is established from the deputy output object of that multimedia device which has the file to be copied to the deputy "data input to digital VTR" object 222. The link data updating means 681 of this object 222 sends a message to the "data input to digital VTR" object 208. As the link updating means 686 of this object 208 is activated, the link data of the

"data input to digital VTR" object 208 are updated.

(0092)

At the same time, the deputy data output object of that multimedia device which has the file to be copied sends a message for updating the link data of the data output object. As the link data are updated, a link is established from the data output object of the multimedia device which has the file to be copied to the "data input to digital VTR" object 208.

(0093)

After this, the data transmitting means of the data output object of the multimedia device which has the file to be copied is activated. The data output object of the multimedia device which has the file to be copied sends a message to the "data input to digital VTR" object. As the data receiving means 1033 and the file writing means 1032 are activated, copying of the file is carried out. In short, when the user gives the copy command or any of the other commands to the deputy data input object and the deputy data output object in the multimedia controller, the deputy data input object and the deputy data output object send the messages to the data input object and the data output object of the main systems of the respective multimedia devices. Thus a link for data communication is established between these two multimedia devices. Concerning the operation of copying data, for example, it is not virtually necessary for the multimedia controller to take direct

participation.

(0094)

As is understandable from the foregoing, according to the prior art, when to make control of the entirety of a system having a plurality of multimedia devices connected to one another, the device drivers or like applications for this purpose had to be previously installed in the controller. According to the invention, however, such a necessity is obviated. So, when a multimedia device is only connected to the LAN, the control panel and the status of the network are automatically displayed in the controller window on the screen. With the help of the windows on the screen, therefore, it becomes easy to turn on and off the power sources of the devices, control the main systems of the devices, and coordinate inputting and outputting of the various signals and data.

(0095)

Another advantage is that of the items the multimedia devices have sent to the control panel of the controller, the ones which are identical in definition to the items the controller has already possessed may be exchanged either in part or all therebetween, depending on the user's taste. As the user interface differs with different makers, it is thus made possible to unitarily rearrange the items.

Yet another advantage is that it becomes possible to execute the control functions from a

controller in the distant place and the functions of accessing to the terminal multimedia devices in a transparent fashion over the LAN.

(0096)

Fig. 30 is a diagram of the structure of a "menu for setting panel view" object of menu class and the relationship between the components of the menu object and the data for describing the object.

(0097)

In Fig. 30, a storage part 1028 accommodates a pointer that goes to a class method table 1430 of the menu class. This table is formed with initializing means 1431 for the menu object, drawing means 1432 for the menu and response means 1433 to selection of the menu. The menu object further includes communicating means 1429 for messages, processing and retrieving means 1434, a storage part 1435 for methods and another storage part 1436 for internal data. The internal data are of an Object ID 1437, the menu state 1438, and the drawing parameters 1439. These data are initialized according to the information from the file 210 for describing the deputy digital VTR object. This file 210 contains a section 211 for describing the "control panel for digital VTR" object. In this section there is a portion related to the menu for setting the panel view. This portion comprises object recognizing data 1440, first object drawing data 1444 for producing a title of the menu, second object drawing data 1448 for producing a first

item in the menu and third object drawing data 1452 for producing a second item in the menu. The object recognizing data 1440 are of the class name 1441 (in this instance, the menu class), an object ID 1442 (in this instance, ID = 2), and a superior object ID (in this instance, ID = 1 of the control panel for the digital VTR). The first object drawing data 1444 are comprised of location and size data 1445, pattern and color data 1446, and an object image 1447, and present the menu title 1462. The second object drawing data 1448 are comprised of location and size data 1449, pattern and color data 1450 and an object image 1451, and represents the first menu item. The third object drawing data 1452 are comprised of location and size data 1453, pattern and color data 1454 and an object image 1455 and represents the second menu item. First object link data 1456 describe a message which is issued when the first menu item 1463 is selected and the destination for the message. A link terminal object ID 1457 represents the ID of the system director object. The message 1458 to send has a text: Create Object with File "Default." Second object link data 1459 describe a message which is issued when the second menu item 1464 is selected and a destination for the message. A link terminal object ID 1460 represents the ID of the system director object. The message to send represents a text: Create Object with File "Custom 1."

(0098)

Fig. 31 is a diagram to explain that the system director object 205, when to read the information for describing the deputy multimedia device object, performs a function of selecting one file out of a plurality of files. In Fig. 31, these files 1465 and 1466 for describing the deputy multimedia device object are named "default" and "custom 1" respectively.

(0099)

Fig. 32 shows the method for the user of selectively setting the control panel. Fig. 33 is a plan view of the selected window on the screen.

(0100)

As shown in Fig. 32, for the "default" display mode, the system director object 205 selects the "default" file 1465 for describing the deputy digital VTR object when the "control panel for digital VTR" object 221 is generated. Therefore, the control panel window 231 for the digital VTR is displayed by default on the screen. At the same time, the menu for the panel view is set in the "Default" option. The user clicks on the "panel view" button to open the menu. Having moved the pointing device, he or she then positions the cursor 230 on top of the "Custom 1" option and clicks on it. Responsive to this, the "menu for setting panel view" object 285 causes the menu option response means 1433 to send the message of "Create object with file 'Custom 1'" to the system director object 205. Responsive to this message, the system director object 205 disposes for once of the

deputy digital VTR object 220 and generates it again by the deputy "multimedia device" object forming means. At this time, the deputy "digital VTR" object written file 1466 named "Custom 1" is read in. By this file, a "control panel for the digital VTR" object 221 is generated. Since the information for an apparent control panel is written in the file 1466 named "Custom 1" as is different from that in the file 1465 named "default," a second or apparent control panel 233 is displayed on the screen a shown in Fig. 33.

(0101)

In the example of Fig. 33, the frame has its corners made round, the buttons are depicted with shadows, and the English labels are conversed to the Japanese ones. Like this, according to the invention, preparation is made of various files for describing deputy objects prepared, so it is possible to meet a wide variety of user's demands.

(0102)

Fig. 34 is plan views showing an example of variation of the layout of control buttons. Though not limited to the multimedia devices, the layouts of the control buttons for appliances generally differ from maker to maker. In a terrible case, one and the same maker has provided different layouts from model to model. This has put the users into confusion. Given that one accustomed oneself to a model and has long used it, when another model is bought in exchange, there is need to

remedy the acquired habit and learn a new management from the beginning. According to the invention, on the other hand, the layout of control buttons can be varied easily by modifying the deputy object written file. It is also possible to give the user a chance to choose a desired one of a plurality of layouts.

Further, the user can change the layout of control buttons in the interactive fashion. In this case, the changed result must be reflected into the deputy object written file by an application of software. Thus, it is also made possible for an individual to customize the control panel freely depending on his or her taste. For another case, either that company who produces, for example, the multimedia controller 1, or a third party, may supply deputy object written files for a wide variety of multimedia devices. If so, it is made possible to rule the differentiation of the layouts of control buttons for all multimedia devices in such a way that the form of Fig. 34(a) is assigned to the maker A, and that of Fig. 34(b) to the maker B by that company or third party.

(0104)

The provision for the multimedia devices with increased capabilities, too, becomes a cause of confusing the user. That is, control members for those functions which the user does not necessitate in the ordinary cases, or, because of his or her appreciation of low level, is unable to use are even displayed in the control panel. Therefore, it often occurs that the control panel becomes

difficult to manage. The present invention can be used even for the purpose of presenting the display with selection of necessary control members according to the user's requirement.

(0105)

Fig. 35 is a plan view of a control panel for a digital VTR having high capabilities displayed on the screen. In Fig. 35, the control panel 276 includes additional functions. Of these, 2767 and 280 are sliders for manually adjusting the level to record sounds at. Level meters 278 and 279 monitor the current levels of recording sound. 281 is a button group for selectively setting image compression modes. A button 282 sets in a compression mode by default. Another button 283 sets in the compression mode of MPEG (for Motion Picture Extraction Grade) according to the international standards. To the users who do not necessitate such high functions, these control members only invite confusion.

(0106)

With the use of the control mode selection object 294 (Fig. 21), the display of the control panel can be changed over between the mode 231 (Fig. 21) which the necessary number of functions is limited to a minimum and the mode 276 (Fig. 35) which permits the user to activate the additional functions. In more detail, the user will first select the advanced button 296 (Fig. 35) by the pointing device. The system director object 205 then reads in that file which the data for displaying

the control panel window 276 for management of the functions including the additional ones are written in, or the file for describing the deputy "control panel for digital VTR" object, and generates again the control panel object for management of the functions including the additional functions. If the additional functions becomes unnecessary, the user selects the "default" button 296 (Fig. 35) by the pointing device. The system director object 205 then reads in that file which the data for displaying the control panel window 231 for management of the necessary minimum number of functions are written in, or the file for describing the deputy "control panel for digital VTR" object and generates again the control panel object for management of the necessary minimum functions.

(0107)

Though the foregoing has been described as using a plurality of files for describing the deputy multimedia device objects in order to permit selection of a plurality of GUIs, it is also possible to write the plurality of GUIs in one file for describing the deputy multimedia device object at the section for writing the deputy multimedia device object therein.

(0108)

(Advantages of the Invention)

As has been described above, according to the invention, when to control all the multimedia devices which are connected to a common system, there is no need

to prepare the device drivers or applications of software for controlling them as were heretofore necessarily installed in the controller. Only when a multimedia device is connected to the LAN, a control panel therefor and the new connection are displayed on the screen of the display of the controller. With this, the user can turn on or off the power source, control the main system of the device, and change over between the input and output modes.

(0109)

The control panel is generated with selection of a plurality of GUIs, thereby making it possible to unify the different user interfaces with different makers. Moreover, it is also possible to select and edit the GUI according to the user's taste. Further, only the necessary control members are displayed depending on the manipulation of the user, thereby making it easier to manage.

(Brief Description of the Drawings)

(Fig. 1)

A diagram showing one form of the logic network of a multimedia controller and multimedia devices.

(Fig. 2)

Diagrams showing physical configurations of connecting a multimedia controller to multimedia devices.

(Fig. 3)

A diagram of the internal structure of an object

embedded multimedia device.

(Fig. 4)

A diagram of the internal structure of an object embedded multimedia controller.

(Fig. 5)

A diagram of a hierarchy of systems in the multimedia controller.

(Fig. 6)

A diagram, of a hierarchy of systems in the multimedia device.

(Fig. 7)

A diagram of a multimedia controller and a multimedia device before they are connected to each other.

(Fig. 8)

A diagram of the multimedia device connected to a LAN.

(Fig. 9)

A diagram of the structure of a common class library.

(Fig. 10)

A diagram of the structure of an object.

(Fig. 11)

A diagram of the structure of a system director object.

(Fig. 12)

A diagram of the structure of a control panel section in a file for describing a deputy object.

(Fig. 13)

A diagram of the structure of a data input/output object section in the file for describing a deputy object.

(Fig. 14)

A diagram of an object embedded digital VTR before connected to the multimedia controller.

(Fig. 15)

A diagram of the structure of a VTR controller object.

(Fig. 16)

A flowchart of the operation that follows the connection of the digital VTR to the LAN.

(Fig. 17)

A plan view of a multimedia controller window on the screen.

(Fig. 18)

A diagram of the object embedded digital VTR connected as a multimedia device to the LAN.

(Fig. 19)

A plan view of an icon for the digital VTR.

(Fig. 20)

A plan view of a control panel window on the screen.

(Fig. 21)

A diagram, partly in plan view, to explain the correspondence of the classes to which the objects belong with the constituent elements of the digital VTR control panel object.

(Fig. 22)

A diagram to explain the production of a play button object.

(Fig. 23)

Flowcharts of the routines for activation of the icon for the digital VTR and for start of a playing operation.

(Fig. 24)

A plan view of a multimedia controller window on the screen after the user has selected the control mode in the icon display for the digital VTR.

(Fig. 25)

A diagram of the relationship between the structure of the deputy "data input to digital VTR" object and the data for describing the object.

(Fig. 26)

A diagram of the relationship between the structure of the deputy "data output from digital VTR" object and the data for describing the object.

(Fig. 27)

A diagram of the structure of the "data input to digital VTR" object.

(Fig. 28)

A diagram of the structure of the "data output from digital VTR" object.

(Fig. 29)

A diagram of the relationship between the structure of the digital VTR control panel object of panel

class and the data for describing the object.

(Fig. 30)

A diagram showing the relationship between the structure of the "control panel for digital VTR" object of the panel class and the data for describing the objects.

(Fig. 31)

A diagram to explain the function of selecting one of a plurality of files when the system director is to read the file for describing the deputy multimedia device object.

(Fig. 32)

A plan view showing the way in which to change over between the control panels.

(Fig. 33)

A plan view of a control panel as obtained after changed over.

(Fig. 34)

A plan view showing an example of variation of the layout of control buttons.

(Fig. 35)

A plan view of the control panel for the digital VTR.

24.04.95

(Name of Document)

Drawings

(Fig. 1)
Logic of Connection of Multimedia Controller with multimedia devices

1: Multimedia Controller

2: Multimedia Device (Digital Camera)
2: Multimedia Device (Printer)
2: Multimedia Device (Digital FAX)
2: Multimedia Device (Digital Copier)
2: Multimedia Device (Digital VTR)
2: Multimedia Device (CD Player)

(Fig. 2)
Physical configurations of connecting Multimedia Controller to Multimedia Devices

a) Daisy Chain Line
1: Multimedia Controller
2: Multimedia Device

b) Star Configuration
1: Multimedia Controller
2: Multimedia Device

c) Multipoint Line
1: Multimedia Controller
2: Multimedia Device

(Fig. 3)
Internal Structure of Object Embedded Multimedia Device

10: external Bus
15: Multimedia Data
16: Driver for Mechanical System
17: Mechanisms & Motors
18: Driver for Electrical System
19: Electrical Circuits, Indicators & Switches

(Fig. 4)

Internal Structure of Object Embedded Multimedia Controller

25: Multimedia Data Filing
26: Display Controller
27: Display
28: Driver for Electrical System
29: Electrical Circuits, Indicators & Switches
30: External Bus

(Fig. 5)
Hierarchy of Systems in Multimedia Controller

50: Hardware
53: Common Class Library
54: C Function
55: Specific Class Library
56: Application for Controlling Multimedia Devices

(Fig. 6)
Hierarchy of Systems in Multimedia Device

57: Hardware
59: Specific Class Library
60: C Function
61: Application for Multimedia Device

(Fig. 7)
1: Multimedia Controller
    205: System Director Object

2: Multimedia Device
    1061: File for Describing Deputy Multimedia Device Object
        1062: Section for Describing "Control Panel for Multimedia Device" Object
        1063: Section for Describing Deputy "Data I/O of Multimedia Device" Object
    1064: Multimedia Device Object
        1065: "Controller in Multimedia Device" Object
        1066: "Data Input to Multimedia Device" Object
        1067: "Data Output from Multimedia Device" Object

(Fig. 8)
1: Multimedia Controller
    205: System Director Object

    1068: Deputy Multimedia Device Object
        1069: "Control Panel for Multimedia Device" Object
        1070: Deputy "Data Input to Multimedia device" Object
        1071: Deputy "Data Output from Multimedia Device" Object

2: Multimedia Device
    1064: Multimedia Device Object
        1065: "Controller in Multimedia Device" Object
        1066: "Data Input to Multimedia Device" Object
        1067: "Data Output from Multimedia Device" Object

(Fig. 9)
1081: Class Library
- 1079: First Class
  - 1080: Class Definition Part
  - 1081: Class Method Table
  - 1082: Code Part
    - 1083: 1st Function Code
    - 1084: k-th Function Code
- 1085: p-th Class

(Fig. 10)
234: Object
- 235: Portion for Internal Data
  - 236: 1st Internal Data
  - 237: 2nd Internal Data
  - 238: n-th Internal Data
- 239: Portion for Methods
  - 240: 1st Data Processing Means
  - 241: 2nd Data Processing Means
  - 242: m-th Data Processing Means
- 243: Class Method Table
  - 240: 1st Data Processing Means
  - 241: 2nd Data Processing Means
  - 242: 3rd Data Processing Means
- 244: Portion for Accommodating Pointer for Class Method Table
- 245: Message Communicating Means; Message
- 246: Processing & Retrieving Means

External Data

(Fig. 11)
205: System Director Object
- 342: Processing & Retrieving Means
- 1061: File for Describing Deputy Multimedia Device Object
- 1072: Portion for Accommodating Pointer for Class Method Table
- 1073: Class Method Table for System Director
  - 1047: Deputy Multimedia Device Object Forming means
  - 343: Input/output Data Checking Means
  - 380: Application Object Forming Means
- 1074: Message Communicating Means; Message
- 1075: Portion for Methods
  - 1047: Deputy Multimedia Device Object Forming means
  - 343: Input/Output Data Checking Means
  - 380: Application Object Forming Means
- 1076: Portion for Internal Data
  - 1077: Object ID
  - 1078: Object Registration Data
  - 344: Current Link Tracking Data

(Fig. 12)
247: Section for Describing Control Panel Object
  248: 1st Database for Describing Object
    250: Data for Recognizing Object
      251: Class Name
      252: Object ID
      253: Superior Object ID
    254: Data for Drawing Object
      255: 1st Data for Drawing Object
        256: Data for Location & Size
        257: Data for Pattern & Color
        258: Object Image
      259: j-th Data for Drawing Object
    260: Data for Object Link
      261: 1st Data for Object Link
        262: Relational Object ID
        263: Message for Transmission
      264: k-th Data for Object Link
  249: i-th Database for Describing Object

(Fig. 13)
650: File for Describing Deputy Data I/O Object
  651: 1st Data for Deputy Data Input Object
    652: Object ID
    653: Relational Data Input Object ID
    654: Compatible File Type List
  655: m-th Data for Deputy Data Input Object
  659: 1st Data for Deputy Data Output Object
    660: Object ID
    661: Relational Data Output Object ID
    662: Compatible File Type List
  663: n-th Data for Deputy Data Output Object

(Fig. 14)
1: Multimedia Controller
  205: System Director Object

203: Digital VTR
  206: Digital VTR Object
    207: Digital VTR Controller Object
    208: "Data Input to Digital VTR" Object
    209: "Data Output from Digital VTR" Object

  210: File for Describing Deputy Digital VTR Object
    211: Section for Describing "Control Panel for Digital VTR" Object
    212: Section for Describing Deputy "Data Output from Digital VTR" Object

(Fig. 15)
207: Digital VTR Controller Object
  1009: Portion for Accommodating Pointer for Class Method Table
  1010: Message Communicating Means
  1011: Processing & Retrieving Means
  1012: Portion for Methods

1019: Reproducing Means
1020: Recording Means
1015: Portion for Internal Data
204: Object ID
1016: Tape Running Speed
1017: Current Footage
1018: Class Method Table for Controller Class
1019: Reproducing Means
1020: Recording Means

(Fig. 16)
636: Connect Digital VTR to Network
637: System Director Object Detects When Digital VTR is Connected
638: System Director Object issues Device ID to Digital VTR
639: System Director Object Loads File for describing Deputy Digital VTR Object
640: System Director Object Generates Deputy Digital VTR Object Based on Information from the File
641: Deputy Digital VTR Object Displays Icon (Mini Panel) for Digital VTR in System Controller Window
642: Wait for Instruction from User

(Fig. 17)

(Fig. 18)
1: Multimedia Controller
205: System Director Object
220: Deputy Digital VTR Object
221: Deputy "Control Panel for Digital VTR" Object
222: Deputy "Data Input to Digital VTR" Object
223: Deputy "Data Output from Digital VTR" Object
203: Digital VTR
206: Digital VTR Object
207: Digital VTR Controller Object
208: "Data Input to Digital VTR" Object
209: "Data Output from Digital VTR" Object

(Fig. 19)
(Fig. 20)
(Fig. 21)
221: Panel Class: Control Panel for Digital VTR
285: Menu Class: Panel View Option Menu
286: Form Class: Timer Counter
288: Button Class: Rewind Button
289: Button Class: Reverse Play Button
290: Button Class: Pause Button
291: Button Class: Fast Feed Button
292: Button Class: Stop Button
293: Button Class: Recording Button
294: Button Group Class: Selection of Control Modes
295: Radio Button Class: Default Button
296: Radio Button Class: Advanced Button

(Fig. 22)

612: Play Button Object
- 613: Portion for Accommodating Pointer for Class Method Table
- 614: Message Communicating Means
- 615: Processing & Retrieving Means
- 616: Portion for Methods
  - 626: Button Initializing Means
  - 627: Button Drawing Means
  - 628: Click Response Means
- 620: Portion for Internal Data
  - 621: Object ID
  - 622: Data for Button State
  - 623: Drawing Parameter
  - 624: Link Data
- 625: Button Class Method Table for Button Class
  - 626: Button Initializing Means
  - 627: Button Drawing Means
  - 628: Click Response Means

625: Button When Not Pressed
626: Button When Pressed
297: Data for Object Recognition
- 298: Class Name: Button Class
- 299: Object ID: ID=7 Play Button
- 300: Super Object ID; ID=1 VTR Control Panel

601: 1st Data for Drawing Object
- 602: Data for Location & Size
- 603: Data for Pattern & Color
- 604: Object Image

605: 2nd Data for Drawing Object
- 606: Data for Location & Size
- 607: Data for Pattern & Color
- 608: Object Image

609: Object Link Data
- 610: Link Terminal Object ID ; ID of VTR Control Object
- 6121: Message to Transmit; Play

(Fig. 23)
643: User Double Clicks on Icon of Digital VTR
644: "Control Panel for Digital VTR" Object Presents Display of Control Panel for Digital VTR
645: Wait for User's Action
646: Click on Play button
647: "Control Panel for Digital VTR" Object Sends "Play" Message to Digital VTR Controller Object
648: Digital VTR Controller Object Activates Reproducing Means
649: Start Play Mode of Digital VTR

(Fig. 24)

(Fig. 25)

222: Deputy "Data Input to Digital VTR" Object
- 668: Portion for Accommodating Pointer for Class Method Table
- 669: Message Communicating Means
- 670: Processing & Retrieving Means
- 671: Portion for Methods
  - 680: Means for Initializing Deputy Data Input Object
  - 681: Link Data Updating Means
  - 678: Compatible File Type Replying Means
- 674: Portion for Internal Data
  - 675: Object ID
  - 676: Relational Data Input Object ID
  - 677: Compatible File Types
  - 1006: Link Data
- 679: Class Method Table For Deputy Data Input Class
  - 680: Means for Initializing Deputy Data Input Object
  - 681: Link Data Updating Means
  - 678: Compatible File Type Replying Means
- 682: Deputy Data Input Object Data
  - 683: Object ID
  - 684: Relational Data Input Object ID
  - 685: Compatible File Type List

(Fig. 26)

223: Deputy "Data Output from Digital VTR" Object
- 690: Portion for Accommodating Pointer for Class Method Table
- 691: Message Communicating Means
- 692: Processing & Retrieving Means
- 693: Portion for Methods
  - 694: Means for Initializing Deputy Data Input Object
  - 695: Means for Updating Link Data
  - 700: Means for replying Compatible File Type
- 696: Portion for Internal Data
  - 697: Object ID
  - 698: Relational Data Output Object ID
  - 699: Compatible File Types
  - 688: Link Data
- 1048: Class Method Table for Deputy Data Output Class
  - 694: Means for Initializing Deputy Data Output Object
  - 695: Means for Sending Data Input Command
  - 700: Means for replying Compatible File Type
- 1001: Deputy Output Object Data
  - 1002: Object ID
  - 1003: Relational Data Output Object ID
  - 1004: Compatible File Type List

(Fig. 27)
208: "Data Input to Digital VTR" Object
- 1022: Portion for Accommodating Pointer for Class Method Table
- 1023: Message Communicating Means
- 1024: Processing & Retrieving Means
- 1025: Portion for Methods
  - 1032: Means for Writing File
  - 1033: Means for Receiving Data
  - 686: Means for updating Link data
- 1028: Portion for Internal Data
  - 1029: Object ID
  - 1030: Link Data
- 1031: Class Method Table for Data Input Class
  - 1032: Means for Writing File
  - 1033: Means for Receiving Data
  - 686: Means for Updating Link Data

(Fig. 28)
209: "Data Output from Digital VTR" Object
- 1035: Portion for Accommodating Pointer for Class Method Table
- 1036: Message Communicating Means
- 1037: Processing & Retrieving Means
- 1038: Portion for Internal Data
  - 1045: Means for reading Data
  - 1046: Means for Transmitting Data
  - 687: Means for Updating Link Data
- 1041: Portion for Internal Data
  - 1042: Object ID
  - 1043: Link Data
- 1044: Class Method Table for Data Output Class
  - 1045: Means for Reading Data
  - 1046: Means for Transmitting Data
  - 687: Means for Updating Link Data

(Fig. 29)
221: "Control Panel for Digital VTR" Object
- 1401: Portion for Accommodating Pointer for Class Method Table
- 1402: Class Method Table for Panel Class
  - 1403: Means for initializing Panel
  - 1404: Means for Drawing Panel Drawing
  - 1405: Click Response Means
- 1408: Portion for Methods
  - 1403: Means for Initializing Panel
  - 1404: Means for Drawing Panel Drawing
  - 1405: Click Response Means
- 1410: Portion for Internal Data
  - 1411: Object ID
  - 1412: Data for Panel State
  - 1413: Drawing Parameters
- 1414: Data for Object Recognition
  - 1415: Class Name; Button Class
  - 1416: Object ID; ID = 1 Digital VTR Control Panel

1417: Super Object ID
1418: 1st Data for Drawing Object
1419: Data for Location & Size
1420: Data for Pattern & Color
1421: Object Image
1422: 2nd Data for Drawing Object
1423: Data for Location & Size
1424: Data for Pattern & Color
1425: Object Image
1426: Icon Image
1427: Control Panel for Digital VTR (Frame)

(Fig. 30)

285: "Menu for Setting Panel View" Object
1428: Storage Part for Pointer That Goes to Class Method Table
1429: Message Communicating Means
1430: Class Method Table of Menu Class
1431: Menu Initializing Means
1432: Menu Drawing Means
1433: Menu Option Response Means
1434: Processing & Retrieving Means
1435: Method Storage Part
1431: Menu Initializing Means
1432: Menu Drawing Means
1433: Menu Option Response Means
1436: Internal Data Storage Part
1437: Object ID
1438: Menu State Data
1439: Drawing Parameters
1440: Data for Recognizing Object
1441: Class Name; Menu Class; Label
1442: Object ID; ID=2; Menu for Setting Panel View
1443: Superior Object ID; ID=1; Control Panel for Digital VTR
1444: Data for Drawing First Object
1445: Data for Location & Size
1446: Data for Pattern & Color
1447: Object Image
1448: Data for Drawing Second Object
1449: Data for Location & Size
1450: Data for Pattern & Color
1451: Object Image
1452: Data for Drawing Third Object
1453: Data for Location & Size
1454: Data for Pattern & Color
1455: Object Image
1456: Link Data for First Object
1457: Link Terminal Object ID; System Director Object
1458: Message to Send; Create Object With File "Default"
1459: Link Data for Second Object
1460: Link Terminal Object ID; System Director

Object
1461: Message to Send: Create Object with File "Custom 1"

(Fig. 31)
205: System Director Object
1072: Storage Part for Pointer That Goes to Class Method Table
1073: Class Method Table of System Director Class
1047: Deputy Multimedia Device Object Forming Means
343: Input/Output data Checking Means
380: Application Object Forming Means
1074: Message Communicating Means
1075: Method Storage Part
1047: Deputy Multimedia Device Object Forming Means
343: Input/Output Data Checking Means
380: Application Object forming Means
1076: Internal Data Storage Part
1077: Object ID
1078: Object Registration Data
344: Current Link Information Control Data
342: Processing & Retrieving Means
1465: "Deputy Digital VTR Object" Written File "Default"
1466: "Deputy Digital VTR Object" Written File "Custom 1"

(Fig. 32)

(Fig. 33)
Time; Default; Advanced; Control Mode;
Rewind; Backward Play, Pause; Play; Fast Feed
Stop; Record

24.04.95

(Name of Document)

Written Abstract

(Abstract)

(Object)

To realize an environment which, when in controlling multimedia devices, does not necessitate applications of software, device drivers or like special software therefor, and on which the multimedia devices can be transparently manipulated from another controller through the LAN.

(Constitution)

A control system in which one controller controls a plurality of object embeded peripheral devices through a common communication line. The controller and the peripheral devices each have a duplex interface. The data for any related objects to the controls of the functions are previously stored in every one of the peripheral devices. The controller, when connected to a peripheral device via the communication line, loads the object data from this peripheral device to form a corresponding object to the peripheral device. Based on the object data, a control panel for controlling the peripheral device is displayed on the screen.

(Selected Figure) Fig. 21